<u>IN THE SPECIFICATION</u>

Please amend the paragraph beginning on page 3, line 10, as follows:

In another aspect of the invention, said calculating $P(X_0 \ldots X_n \mid S_0 \ldots S_n)$ comprises assigning a state label S_i to each i^{th} byte X_i of the byte sequence so as to maximize the equation:

$$P(X_0 \ldots X_N \mid S_0 \ldots S_N) = P_0(S_0)^{\left[\prod_{i=1}^{N} \bar{A}(S_i \mid S_{i-1})\right]\left[\prod_{i=0}^{N} \bar{B}(X_i \mid S_i)\right]}$$

wherein $P_0(S_0)$ is the initial distribution of states; $\bar{A}(S_i \mid S_{i-1})$ is a "state-to-state" ~~transmission~~ <u>transition</u> matrix; and $\bar{B}(X_i \mid S_i)$ is a "byte-from-state" matrix of the probabilities of generating a byte value X_i given a state S_i.

Please amend the paragraph beginning on page 4, line 7, as follows:

In another aspect of the invention,

$$\bar{B}(X_i \mid S_i) = \begin{bmatrix} h_1(X_i = 1) & \cdots & h_s(X_i = 1) & \cdots & h_\sigma(X_i = 1) \\ \vdots & \ddots & \vdots & & \vdots \\ h_1(X_i = x_1) & & h_s(X_i = x_1) & & h_\sigma(X_i = x_1) \\ h_1(X_i = x_1 + 1) & & \varepsilon_1(X_i = x_1 + 1) & & h_\sigma(X_i = x_1 + 1) \\ \vdots & & \vdots & & \vdots \\ h_1(X_i = x_r) & & \varepsilon_1(X_i = x_r) & & h_\sigma(X_i = x_r) \\ h_1(X_i = x_r + 1) & & \varepsilon_r(X_i = x_r + 1) & \ddots & h_\sigma(X_i = x_r) \\ \vdots & & \vdots & \ddots & \vdots \\ h_1(X_i = x_r = 255) & & \varepsilon_r(X_i = x_r = 255) & & h_\sigma(X_i = 255) \end{bmatrix}$$

where $h_S(X_i)$ are histogram functions of the σ states and [[$\aa_j$]] <u>$\varepsilon_j(X_i)$</u> are probabilities of associating noise with the noise state for bytes within $r+1$ ranges of byte values X_i.

Please amend the paragraph beginning on page 5, line 1, as follows:

In another aspect of the invention,

$$
\overline{B}(X_i \mid S_i) = \begin{bmatrix}
h_A(X_i = 1) & 0 & 0 \\
\vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots \\
h_A(X_i = 127) & 0 & 0 \\
\varepsilon_1(X_i = 128) & 0 & 0 \\
\vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots \\
\varepsilon_1(X_i = 160) & 0 & 0 \\
\varepsilon_2(X_i = 161) & h_1(X_i = 161) & h_2(X_i = 161) \\
\vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots \\
\varepsilon_2(X_i = 254) & h_1(X_i = 254) & h_2(X_i = 254) \\
\varepsilon_3(X_i = 255) & 0 & 0
\end{bmatrix}
$$

where $h_s(X_i)$ are histogram functions of the states, and [[å$_j$]] $\underline{\varepsilon_j}(X_i)$ are probabilities of associating noise with the ASCII state within a plurality of X_i ranges for a three-state byte sequence.

Please amend the paragraph beginning on page 5, line 6, as follows:

Another aspect of the invention further comprises, providing a valid three-state byte sequence having an ASCII state and comprising valid ASCII and two-byte characters, computing an ASCII histogram $h_A(X_i)$ by a method comprising sampling valid ASCII text so as to measure the frequency of occurrence of each byte value; computing an unnormalized histogram of said sampling over the ASCII range of X_i; and normalizing said unnormalized histogram such that the entire column of the matrix containing said ASCII histogram ~~said~~ sums to 1, and computing a first-byte histogram $h_1(X_i)$ by sampling valid two-byte text and computing the unnormalized first-byte histogram over the odd bytes, and normalizing said first-byte histogram such that the entire column of the matrix containing said first-byte histogram sums to 1, and computing a second-byte histogram $h_2(X_i)$ by sampling valid two-byte text and computing the unnormalized second-byte histogram over the odd bytes, and normalizing said second-byte histogram such that the entire column of the matrix containing said second-byte histogram sums to 1.

Please amend the paragraph beginning on page 6, line 18, as follows:

Disclosed is a method of validating a byte sequence having a plurality of states including an ASCII state, the method comprising selecting the ASCII state as the noise state generating a most probable state sequence for the byte sequence by a method comprising: calculating $P(X_0 \ldots X_n \mid S_0 \ldots S_n)$, representing the conditional probabilities of said byte sequence given a state sequence, wherein said calculating $P(X_0 \ldots X_n \mid S_0 \ldots S_n)$ comprises assigning a state label S_i to each i^{th} byte X_i of the byte sequence so as to maximize the equation:

$$P(X_0 \ldots X_N \mid S_0 \ldots S_N) = P_0(S_0)\left[\prod_{i=1}^{N} \bar{A}(S_i \mid S_{i-1})\right]\left[\prod_{i=0}^{N} \bar{B}(X_i \mid S_i)\right]$$

wherein $P_0(S_0)$ is the initial distribution of states; $\bar{A}(S_i \mid S_{i-1})$ is a "state-to-state" ~~transmission~~ transition matrix; and $\bar{B}(X_i \mid S_i)$ is a "byte-from-state" matrix of the probabilities of generating a byte value X_i given a state S_I; and utilizing said state sequence to identify all noise in the byte sequence, localizing said noise in said noise states, and deleting said noise from the byte sequence.

Please amend the paragraph beginning on page 8, line 1, as follows:

Figure 2 depicts a typical Markov model for allowable state sequences for mixed double-byte and ASCII sequences, such as GB-type byte sequences. The state of a byte in this example can be one of three, namely an ASCII character (state A), a first byte of a two-byte character state (state GB1), or a second byte of a two-byte character state (state GB2). The states are designated by the user, dependent upon the noise he wishes to detect, which in this example is invalid character codes. As can be seen in the diagram, a single-byte ASCII character (state A) can be followed by another ASCII character (state A) or by the first byte (state GB1) of a double-byte GB-type character, but never can an ASCII character (state A) be followed by the second byte (state GB2) of a double-byte GB-type character. This is shown by the directions of the arrows leading toward and away from the ASCII state A.

Likewise, a first GB byte (state GB1) may be followed by a second byte (state GB2), but never by an ASCII character (state A); and a second byte (state GB2) may be followed by an ASCII character (state A), or by a first byte (state GB1), but never by a ~~first~~ second byte (state GB~~1~~2). A violation of these rules is not permitted in the state string generated by the state sequence labeler 100 of Figure 1 and this is mathematically guaranteed by the zero entries in the "state-to-state" ~~transmission~~ transition matrix $\bar{A}(S_i \mid S_{i-1})$ of Equation 4c, more fully described below.

Please amend the paragraph beginning on page 9, line 13, as follows:

We may then model the conditional probabilities of the byte sequence of Equation 1 given the state sequence of Equation 2 as:

$$P(X_0 \ldots X_N \mid S_0 \ldots S_N) = P_0(S_0)\left[\prod_{i=1}^{N} \bar{A}(S_i \mid S_{i-1})\right]\left[\prod_{i=0}^{N} \bar{B}(X_i \mid S_i)\right] \qquad (3)$$

where $P_0(S_0)$ is the initial distribution of states, namely $P_0(A) = P_0(1) = P_0(2) = 1/3$; $\bar{A}(S_i \mid S_{i-1})$ is the state-to-state ~~transmission~~ transition matrix and will have the properties of the Markov model being utilized; $\bar{B}(X_i \mid S_i)$ is the "byte-from-state" matrix of the probabilities of generating a byte value X_i given a state S_i.